Silicon Turnkey eXpress

Original Design Manufacturer

PRELIMINARY

Dual-460GT AMC Card

"Arches"

Reference Design Kit

User's Guide

Technical support: archessupport@amcc.com

Silicon Turnkey Express
749 Miner Road • Highland Heights, Ohio 44143
Phone (440) 461-4700 • (800) 827-2650

Revision History

Rev	Date	Comments
1.0	Nov 25, 2008	Initial Preliminary Release

WARNING:	This document is preliminary. It may contain errors and incomplete data. Contact archessupport@amcc,com for updates

Support:

Additional support information may be found at
www.amcc.com/Embedded/Downloads/Arches
or email archessupport@amcc.com .

NOTICES

The following information is intended to alert the user to possible dangers and important information contained within this guide. The "**WARNINGS**", "**CAUTIONS**" and "**NOTES**" do not eliminate these dangers. Close attention to the information supplied along with "common sense" operation is the major accident prevention measure.

WARNING:	*Failure to follow this warning may result in bodily injury.*

CAUTION:	*Failure to follow this caution may result in possible damage to the board.*

NOTE:	*Failure to follow this note may result in improper results from the board.*

Reference Websites

Below is a list of websites that can be used to obtain additional information and details that may not be fully provided in this manual.

AMCC ..... www.amcc.com
Abatron JTAG Emulator……. www.ultsol.com/mfgs_emul_abtr.htm
Altera Byteblaster..... www.altera.com
denx Software Engineering…… www.denx.de
ENEA…… www.enea.com
RapidFET…….. www.fetcorp.com
RIO Test Labs…….…….. www.rio-lab.com
PICMG…….……..…… www.picmg.org

Table of Contents

List of Figures

Appendix

General Description

The Arches AMC card is a reference design powered by two PPC460GT processors. The reference design allows evaluation of the PPC460GT processor in a dual architecture in an Advanced Mezzanine Card (AMC). The Arches AMC can also be used as a target system in an ATCA or MicroTCA design. Access to both processors is provided through a variety of buses including JTAG, 1G Ethernet, PCI Express and Serial RapidIO. Each processor has its own resource for memory, directly communicates via Ethernet or PCI Express and can be individually configured through an on board CPLD. The Arches AMC includes a complete IPMI control system for machine configuration.

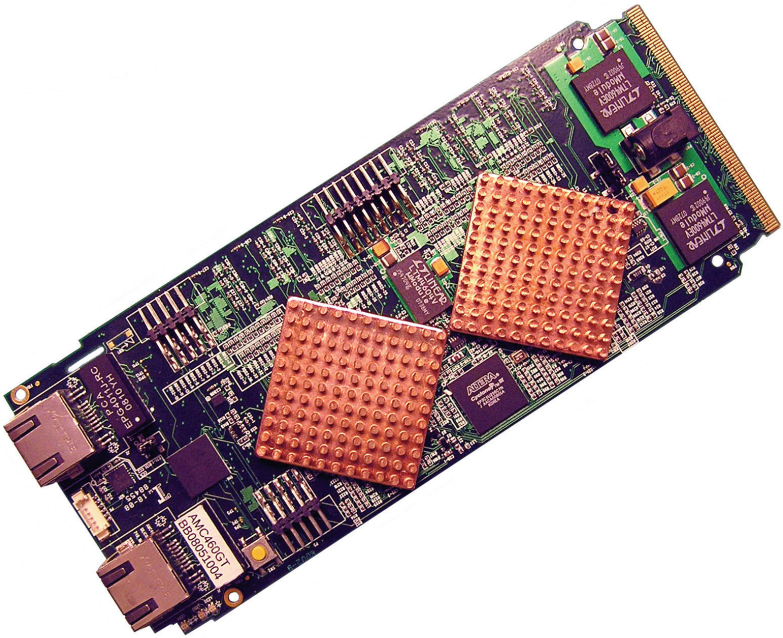

Figure 1 – General Picture

1.1 Device Placement and Functions

This section provides a description of the connectors, jumpers, switches and main components of the board. Refer to Figure 2 and 3 for the location of the devices referenced below.

Additional descriptions of the functionality of switches, jumpers, and LEDs along with their recommended settings will be found in Section 3.

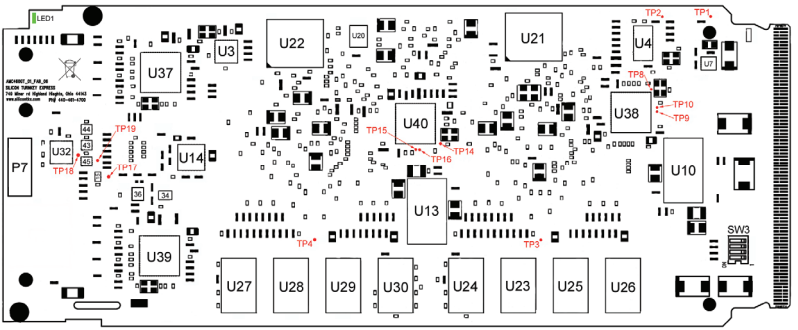

Figure 2 – Top Board Layout

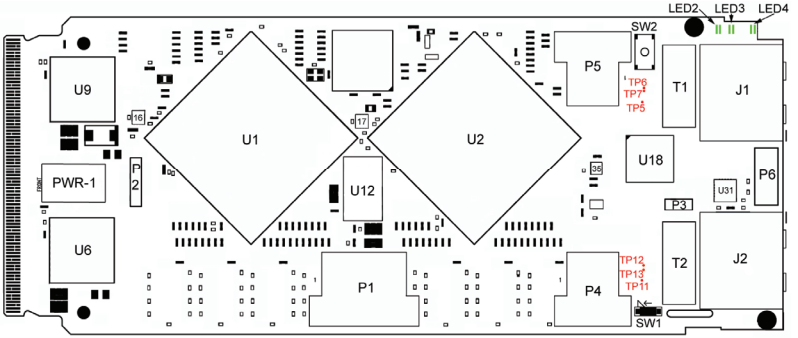

Figure 3 – Bottom Board Layout

1.1.01 J1 – RJ45 10/100/1000 Base-T

J1 is a 1 Gigabyte Ethernet for CPU0.

1.1.02 J2 – RJ45 10/100/1000 Base-T

J1 is a 1 Gigabyte Ethernet for CPU1.

1.1.04 LED 1

LED1 is used by the IPMI (user provided).

1.1.05 LED 2 and 3

LEDs are bi-color (Green & Yellow) and are user defined. See CPLD register 4 for details.

1.1.06 LED 4

LED 4 is bi-color (Red & Green) and is user defined. See CPLD register 4 for details.

1.1.07 P1 – JTAG Debug Port

P1 is a 16 pin port used to debug either CPU0 or CPU1.

1.1.08 P2 – CPU Halt Selector

P2 is a 4 pin header used to halt either or both CPUs for debugging. Use a jumper for selection as follows:

Jumper	CPU
1-2	0
3-4	1
1-2 3-4	Both

1.1.09 P3 – UART Selection

P3 is a 2 pin header used to select either the MMC UART or CPU1 UART at connector P7.

Open = CPU1
Jumped = MMC

1.1.10 P4 – CPLD JTAG Port

P4 is a 10 pin header used to program and debug the CPLD, confirming to the industry-standard pinout used by probe manufacturers such as Abatron, Green Hills, Lauterbach, Wind River and others.

1.1.11 P5 – FPGA Config. Device JTAG

P5 is a 10 pin header used to configure and debug the FPGA. A standard Altera Byteblastter can be used with this connector.

1.1.12 P6 – CPU0 UART Connector

P6 is a standard DB9 connector used to access CPU0's UART.

1.1.13 P7 – UART for CPU1 or MMC Connector

P7 is a standard DB9 connector and is used with jumper P5 to select either CPU1's UART or the MMC's UART.

1.1.14 PWR1 – External Power Plug

PWR1 is used with an external wall 'cube' power supply for standalone operation, such as the 12V power adapter included in the kit.

WARNING:	*DO NOT USE EXTERNAL POWER WHEN AMC EDGE CONNECTOR IS POWERED.*

1.1.15 SW1 – AMC Ejection Switch

SW1 is actuated by the front panel ejection leveler and provides an input to by the IPMI system (user provided).

1.1.16 SW2 – Master Reset

SW2 is a single button switch which provides a power on reset to the module.

1.1.17 SW3 – Configuration Switch

SW3 is a four position switch that set the boot mode for the CPUs.

1.2 Test Points

TP	Function
01	ESD DISCHARGE STRIP
02	ESD DISCHARGE STRIP
03	U1D, PIN AM33/34, DDR-1 Mem DCF
04	U2D, PIN AM33/34, DDR-2 Mem DCF
05	U37 (88E1112), PIN 34, TSTPT
06	U37 (88E1112), PIN 25, HSDAC_P
07	U37 (88E1112), PIN 26, HSDAC_N
08	U38 (88E1112), PIN 34, TSTPT
09	U38 (88E1112), PIN 25, HSDAC_P
10	U38 (88E1112), PIN 26, HSDAC_N
11	U39 (88E1112), PIN 34, TSTPT
12	U39 (88E1112), PIN 25, HSDAC_P
13	U39 (88E1112), PIN 26, HSDAC_N
14	U40 (88E1112), PIN 34, TSTPT
15	U40 (88E1112), PIN 25, HSDAC_P
16	U40 (88E1112), PIN 26, HSDAC_N
17	U45 (74LVC2G66) PIN 2, 1B
18	U45 (74LVC2G66) PIN 1, 1A
19	U45 (74LVC2G66) PIN 7, 1CNT

Figure 4 – Test Points

2.0 Hardware Design & Architecture

2.1 General Description

Features

- Two PowerPC 460GT 1.0GHz CPUs
- CPUs interconnected via
  - Shared memory
  - GigE
  - PCI-express, x1
- Two 512MB DDR2 subsystems
- Two 64MB NOR FLASH
- Two UART Ports
- Two 10/100/1000 Ethernet RJ45 Ports
- Two EEPROM (up to 64KB) via I2C
- Two Temperature Sensor via I2C
- Four 10/100/1000 Ethernet AMC Ports
- One x4 sRIO AMC Port
- One x1/x4 PCI-e/sRIO AMC Port
- Management Module Controller Support
- Shared JTAG Connector
- Linux 2.6
- Usable Stand-Alone or ATCA Chassis
- Full Compliance:
  - AMC.2 E2 (GE)
  - AMC.1 (PCI-e)
  - AMC.4 (sRIO)
  - SCOPE

2.2 Physical Description

Board Size.....180mm x 74mm
Power Requirement…….. 12vDC @ 40watts
Operating Temperature:
Standard…… 0-50^{0}C
Extended……TBD
Weight.....400g
RoHS..... Compliant

2.3 Arches Block Diagram

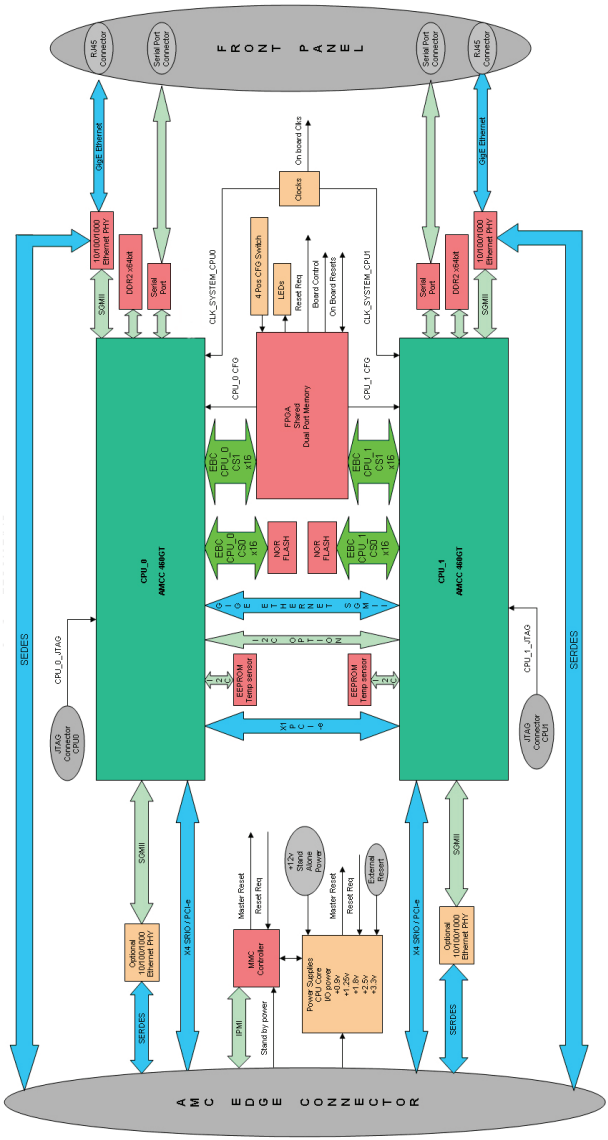

Figure 5 - Arches Block Diagram

3.0 Control and Configuration

3.1 Switch Settings

This section provides a brief description of the functionality and recommended settings for the switches.

Refer to Figure 2 in Section 1 for the locations of these switches.

3.1.1 SW1 – AMC Ejection

This switch provides information to the IPMI that allows the AMC to be ejected from the AMC connector. This function is NOT implemented and is the responsibility of the user.

3.1.2 SW2 – Reset Switch

This is a push button switch when in the stand alone mode provides a Power On Reset to the CPLD which will reset both CPUs, the MMC and the CPLD.

3.1.3 SW3 – Boot Mode Configuration

This 4 position switch provides inputs to the CPLD that sets board boot options and frequencies.

See Figure 6, Configuration Switch Modes for switch settings.

See Section 5.07, CPU Configuration for more information about the Boot Mode Configurations.

See CPLD Register 1 for additional information.

SW3 Boot Mode Configuration (continue)

Switch Position				Switch Function		
1	2	3	4	Boot Opt	SYSCLK	PCIe_CLK
0	0	0	0	E	66.6MHz	100MHz
0	0	0	1	B	66.6MHz	100MHz
0	0	1	0	G	66.6MHz	100MHz
0	0	1	1	E	66.6MHz	125MHz
0	1	0	0	B	66.6MHz	125MHz
0	1	0	1	G	66.6MHz	125MHz
0	1	1	0	Reserved		
0	1	1	1	Reserved		
1	0	0	0	G	74.9MHz	100MHz
1	0	0	1	G	83.3MHz	100MHz
1	0	1	0	G	99.9MHz	100MHz
1	0	1	1	Reserved		
1	1	0	0	G	74.9MHz	125MHz
1	1	0	1	G	83.3 MHz	125MHz
1	1	1	0	G	83.3MHz	125MHz
1	1	1	1	Reserved		

Figure 6 – Configuration Switch Modes

3.2 Jumpers

This section provides a brief description of the functionality and recommended settings for the jumpers. Refer to Figure 2 in Section 1 for the locations of these jumpers.

3.2.1 P2 – CPU Halt Selector

Default Open

This jumper is to select which CPU that is connected to the COP Debug Port, P1. Positions are CPU0, CPU1 or both.

3.2.2 P3 – UART Selector

Default Open

This jumper is selects either the MMC UART or the CPU1 UART. In the normally open (no jumper) CPU1 UART is selected. When the jumper is installed the MMC UART is connected to UART xx, Px

3.3 LEDs

This section provides a list of functions for the LEDs. Refer to Figure 2 for the locations of these LEDs.

See CPLD Register 7, Addendum C, for more information.

LED	Function	Color
1	AMC Eject	Blue
2	User Defined	Grn/Yel
3	User Defined	Grn/Yel
4	User Defined	Grn/Red

4.0 Schematic

The schematic and basic assembly information in a portable document format for the Arches can be downloaded via www.amcc.com/Embedded/Downloads/Arches.

The Arches AMC design can be customized for optional flexibility and custom interfaces so the embedded systems engineer can obtain a lower overall parts cost using a variety of fixed and user selectable options.

These options inherently are contained in connectors, jumpers and switches on the board.

The schematic provides guidelines for using the already installed as well as user modifiable options available on the present design.

5.0 Circuit Operation

5.01 Overview

The Arches reference design is based on two PPC460GT processors in an AMC form factor. CPU1 is a mirror of CPU0 for local resources (each has its own local bus FLASH, I2C device, UART, Gigabit Ethernet, SRIO/PCI-e and DDR2 SDRAM interface). Both processors interface to an FPGA for shared memory. The processors share clock generators and a single power supply. The processors communicate to each other by Giga-bit Ethernet (SGMII) and a 1x PCI-e bus.

On board interrupts are hard wired to the CPUs. See section 5.08 for complete interrupt information.

Only CPU0 communicates with the Management Module Controller, MMC, via a SPI bus. The Intelligent Platform Management Interface (IPMI) communications to the AMC edge connector is generated by the MMC. The user is responsible to implement the IPMI software. The MMC communicates the following to the FPGA:

Enable power supplies
Take CPUs out of reset
USER LEDs
Ethernet resets
Read geographical address

The CPLD reads the configuration switches and communicates them to the CPUs and sets clock configuration to the clock generators. The CPLD drives the signals, but receives the information from the MMC across the MMC I2C bus.

5.02 Power Supplies

Main power is provided from either the AMC edge connector or the +12V barrel connector (PWR1). Arches will auto-detect main power and properly power the module. PWR1 has been positioned to prevent its use while the card is connected to an AMC edge connector.

WARNING:	*Only one power source can be used at a time or damage will occur to the board. Failure to follow this warning may result in bodily injury.*

When +12V is applied to PWR1, stand alone mode, the signal STD_ALONE_Enable is allowed to go high which in turn will turn on +3.3V_stby regulator U7. In turn U7 provides standby power to the FPGA, and if used the MMC, which is vital to the board's initialization.

The main power supplies are enabled by the FPGA. The FPGA initializes the power supplies when in the AMC mode. The FPGA by-passes the MMC communication in the stand alone mode and enable the power supplies after an initialization cycle of about 300 ms. All of the power supplies provide a "power good" signal to the FPGA. If any of the power supplies fail, this signal to the FPGA will cause Arches to shut down immediately.

Power supply rails include: +3.3v_stby; +3.3v; +1.25v; +2.5v; +1.8v; and +0.9v.

Estimated Power Usage:

Device	Qty	3.3v stby	3.3v	2.5v	1.8v	0.9v	1.2v	1.25v	Total
AMCC 460GT_0	1		0.5000	0.5000				4.20	8.15000
AMCC 460GT_1	1		0.5000	0.5000				4.20	8.15000
U15	1	0.050							0.16500
U17	1		0.0500						0.16500
U18	1	0.050							0.16500
LEDs	1	0.010							0.03300
LEDs	7		0.0100						0.03300
U19 & U20	2	0.016							0.05280
U3 & U4	2		0.1000						0.33000
ICS8402	1		0.1530						0.50490
ICS844246	1		0.2610						0.86130
PSs	3		2.5000						8.25000
UARTS	2		0.0200						0.06600
UARTS	1	0.020							0.06600
U24	1		0.0150						0.04950
U45	1		0.1000	0l05			0.25		0.75500
U25 & U26	2		0.1800						0.59400
DDR2 CPU_0	4				1.40	1.40			3.78000
DDR2 CPU_1	4				1.40	1.40			3.78000
EEPROM	2		0.0030						0.00990
Temp Sensor	2		0.0012						0.00396
Enet Phy	4			0.2470					0.99430
Totals		0.162	0.8024	2.0380	11.20	11.20	1.51	8.40	37.11000

Figure 7 - Estimated Power Usage Table

5.03 JTAG Port

The JTAG port is configured as a single connector, P1, with both processors daisy chained together with the CPU0, U1, as the first in the chain.

When using an Abatron BDI2000/3000 probe, the configuration script has to be modified and a jumper, P2, placed depending on which CPU is be interrogated.

CPU0: Jumper P2 pins 1 and 2 (Halts CPU1)

Script: SCANSUCC 1 0x07

CPU1: Jumper P2 pins 3 and 4 (Halts CPU0)

Script: SCANPRED 1 0x07

5.04 Clocks

The main on board system clocks are controlled by U3. The system clock controls the CPUs, the FPGA, and the CPLD. The output clocks are set by the CPLD driving SYS_CLK_SEL1…3 signals to U3 and be set to the frequencies as follows:

66.66667MHz	SYS_CLK_SEL3…1 = 000
74.99999MHz	SYS_CLK_SEL3…1 = 001
83.33333MHz	SYS_CLK_SEL3…1 = 010
99.99999MHz	SYS_CLK_SEL3…1 = 100

The PCI-express clock is controlled by U4. The CPLD controls U4 with the CLK_N_SEL1…0 signal and can be set to the following frequencies:

125MHZ	CLK_N_SEL1…0 = 01
100MHZ	CLK_N_SEL1…0 = 10

5.04 Clocks (continued)

Y3 is a stand alone oscillator providing a 50MHz clock to the FPGA. The FPGA generates each CPU's TMRCLK independent from each other. Currently the FPGA does not manipulate this clock, but the FPGA can change it, if necessary.

Y4 is a stand alone oscillator providing a 50Mz clock to each CPU for Ethernet management.

5.05 Resets

Arches resets are generated from the AMC connector in the AMC Mode or the FGPA in the stand alone mode.

In the stand alone mode the MCLR# signal is the same as the AMCE_ENABLE# and is pulled high after an initial time out period of about 300 ms. This allows the use of the AMC connector for debug of the MMC. A custom cable is required for the AMC connector. In this mode the CPLD will control the board bring up based on the power being enabled and the initial time out period of about 300 ms. Switch SW2 will provide a Power-On-Reset to the CPLD which resets both CPUs, the MMC, and the CPLD.

In the AMC mode the MCLR# signal is the opposite of the AMCE_ENABLE# signal. In this mode after the initial release of the AMCE_ENABLE# signal the MMC communicates to the CPLD via the MMC I2C bus. The MMC writes a specific value to the CPLD boot register (see Appendix C, Register 5).

The MMC monitors all the voltage rails and should one voltage fail will reset the AMC board. This event powers down all power supplies except the standby power.

5.06 Module Management Controller, MMC

The Module Management Controller, also known as the Baseboard Management Controller (BMC) is the heart of the IPMI based subsystem. For more information about IPMI see 'Making IPMI Work in ATCA Designs' by Steve Rokov, OSA Technologies, dated November 10, 2004.

Arches' MMC uses a PIC24FJ64GA002. The user is responsible for the development of software for this processor, such as, IPMI software. A jumper, Pxx, has been installed to by-pass all MMC functionality. Functions required by Arches by the MMC will be handled by the CPLD/FPGA.

Neither AMCC nor STx provides support for user-developed MMC software. However an example of minimum MMC functionality would include communicating to the TCA or uTCA blade via the I2C bus AMCE_SDA_L, serial data signal and the AMCE_SCL, serial clock signal. The MMC would also use the I2C to communicate to the CPLD (in the stand alone mode this is not necessary, if the default values are acceptable). The MMC is also is responsible for controlling: monitoring reset switches and generating reset signals to the CPLD (the MMC_CLKOUT signal was changed to this function); LED1 (blue); hot swap switch, SW1; board resets via SW2; MMC UART; power supply voltage monitoring; communication to the main CPU, CPU0, U1; and communicating with temperature sensors for each CPU via I2C, U16 and U17.

Factory programming uses a custom cable connected to the AMC edge connector for the following signals:

AMCE_SDA_L	serial data signal
AMCE_SCL	serial clock signal
AMCE_ENABLE#	
+3.3v_stby	
GND	
Bootloader	Microchip updater

5.07 CPU Configuration

The CPUs' initial configuration is controlled by the CPLD and configuration switch SW3. The CPLD provides the following signals:

CPU0_boot_CFG0…1	Bootstrap signal to CPU0
CPU1_boot_CFG0…1	Bootstrap signal to CPU1
SYS_CLK_SEL1…3	Main system clock frequency
CLK_N_SEL0…1	PCI-e clock frequency

See the CPLD register 2, Appendix C, for complete configurations.

5.08 Interrupts

All Arches interrupts are directly connected to the processors.

The following list the IRQs and their signals.

Interrupt	Driven From
CPU0_IRQ00	ETH0_INT
CPU0_IRQ01	ETH1_INT
CPU0_IRQ02	CPU0_TEMP_IRQ#
CPU1_IRQ00	ETH2_INT
CPU1_IRQ01	ETH3_INT
CPU1_IRQ02	CPU1_TEMP_IRQ#

5.09 DDR2 SDRAM Memory

Each CPU has a dedicated DDR2 memory bus. Each bus is 64 bit, has a single bank of 512 MB with a maximum speed of 400 MHz, and no Error Checking and Correction. Each bus is directly controlled by each CPU's SDRAM controller.

5.10 NOR FLASH

Each CPU has dedicated 64 MByte of NOR FLASH on the x16 bit bus. The FLASH uses chip select PerCS0#.

5.11 FPGA

The FPGA requires a configuration device. The configuration device is programmed with an Altera Byteblaster II cable or equivalent plugged in to P5. It is recommended to use the Active Programming setting within the Altera Quartus software.

The main function of the FPGA is to provide 2 Mbytes of shared memory via the x 16bit bus to both processors. It provides address decodes as follows:

Address 12-15 (b0000) is FPGA board control and status registers.

Address 12-15 (b1000) is CPLD board control and status registers (Only CPU_0 can access the CPLD).

Address 12-15 (b1100) is shared memory

See FPGA registers for additional board address decodes and board control and status registers.

The FPGA provides clocks CLK_TMRCLK0 and CLK_TMRCLK1. These are independent clocks based on the input clock CLK_TMRCLK_OSC and default to 50 MHz. These clocks can be modified by the FPGA.

All FPGA code can be modified to suite specific needs.

5.12 I^2C

Each CPU has an EEPROM (address A8) and temperature sensor (address 94) on the I2C0 bus.

5.13 UART

Each CPU has a UART port. CPU0 is connected to P6 through a transceiver. CPU1's UART is shared with the MMC controller and is selected with jumper P3. The default of P3 (jumper installed) connects CPU1 to the UART.

The UART connectors (P6 and P7) use a small Molex connector and require a custom cable (one supplied in the RD-460GT-KIT-01) to a standard DB9 connector. See Appendix D for pin assignments of these connectors.

5.14 Gigabit Ethernet

Each CPU has two Gigabit Ethernet connected to the CPU's Ethernet management interface using SGMII. The Ethernet use a Marvel 88E1112 'PHY' for either a 10/100/1000Base-T or 1.25 GHz SERDES. The PHY is configured to take advantage of the Media Detection™ mode for copper or fiber support. The PHY is connected to an RJ45 connector on the front panel for Ethernet support and 1.25GHz SERDES on the AMC edge connector.

NOTE: Only one interface can be used at a time.

The other Ethernet PHY is wired for 1.25GHz SERDES on the AMC edge connector.

Both CPUs are connected directly together (AC Coupled) using the SGMII 0 ports.

5.15 SGMIII Hardware Matrix

Both CPUs are wired exactly the same, except SGMII0 Rx on CPU0 is connected to SGMII0 Tx on CPU1.

EMAC0	CPU0_SGMII 0	CPU1_SGMII 0
EMAC1	SGMII 1	Ethernet PHY address 0000
EMAC2	SGMII 2	Ethernet PHY address 0001

5.16 Serial Rapid I/O / PCI-express

There are two independent PCI Express interfaces compliant to specification 1.1. One interface can be configured as one to four lanes while the other functions as one lane only. The four lane interface shares a high speed SERDES with the Serial Rapid IO (SRIO) interface. Both can be Root or Endpoint Ports.

The CPU's PCIe0 port is used for communicating between processors using the x1 lane PCIe bus. PCIe is serial interface and is AC coupled from TX on CPU0 to RX on CPU1 and RX on CPU0 to TX on CPU1.

The x4 PCIe bus is multiplexed with the x4 SRIO bus and routed to the AMC edge connector. See AMCC 460GT documentation for details.

6.0 U-Boot

6.1 U-Boot Commands

```
?         - alias for 'help'
askenv     - get environment variables from stdin
autoscr   - run script from memory
base      - print or set address offset
bdinfo    - print Board Info structure
boot      - boot default, i.e., run 'bootcmd'
bootd     - boot default, i.e., run 'bootcmd'
bootelf   - Boot from an ELF image in memory
bootm     - boot application image from memory
bootp     - boot image via network using BootP/TFTP protocol
bootstrap - program the I2C bootstrap EEPROM
bootvx    - Boot vxWorks from an ELF image
cmp       - memory compare
coninfo   - print console devices and information
cp        - memory copy
crc32     - checksum calculation
dcache    - enable or disable data cache
dhcp      - invoke DHCP client to obtain IP/boot params
dtt       - Digital Thermometer and Thermostat
echo      - echo args to console
eeprom    - EEPROM sub-system
erase     - erase FLASH memory
exit      - exit script
fdt       - flattened device tree utility commands
flinfo    - print FLASH memory information
getdcr    - Get an AMCC PPC 4xx DCR's value
getidcr   - Get a register value via indirect DCR addressing
go        - start application at address 'addr'
help      - print online help
icache    - enable or disable instruction cache
icrc32    - checksum calculation
iloop     - infinite loop on address range
imd       - i2c memory display
iminfo    - print header information for application image
imls      - list all images found in flash
imm       - i2c memory modify (auto-incrementing)
```

U-Boot Commands (continued)

```
imw         - memory write (fill)
imxtract    - extract a part of a multi-image
inm         - memory modify (constant address)
interrupts  - enable or disable interrupts
iprobe      - probe to discover valid I2C chip addresses
irqinfo     - print information about IRQs
itest       - return true/false on integer compare
loadb       - load binary file over serial line (kermit mode)
loads       - load S-Record file over serial line
loady       - load binary file over serial line (ymodem mode)
loop        - infinite loop on address range
loopw       - infinite write loop on address range
md          - memory display
mdc         - memory display cyclic
mii         - MII utility commands
mm          - memory modify (auto-incrementing)
mtest       - simple RAM test
mw          - memory write (fill)
mwc         - memory write cyclic
nfs         - boot image via network using NFS protocol
nm          - memory modify (constant address)
ping        - send ICMP ECHO_REQUEST to network host
printenv    - print environment variables
protect     - enable or disable FLASH write protection
rarpboot    - boot image via network using RARP/TFTP protocol
reginfo     - print register information
reset       - Perform RESET of the CPU
rio         - list and access RapidIO devices
run         - run commands in an environment variable
saveenv     - save environment variables to persistent storage
setdcr      - Set an AMCC PPC 4xx DCR's value
setenv      - set environment variables
setidcr     - Set a register value via indirect DCR addressing
sleep       - delay execution for some time
test        - minimal test like /bin/sh
tftpboot    - boot image via network using TFTP protocol
version     - print monitor version
```

6.2 U-Boot SRIO Commands

```
rio init <lanes 1 or 4> <hdid>                - initialize RapidIO interface
rio test <cmd> <delay in mseconds>            - run test (cmd=link,...)
rio reset <cmd>                               - reset (cmd=rsync,tsync)
rio status                                    - display RapidIO status
rio dump  <reg> <did>                         - print registers
                                                (reg=sdr|cfg|reg|mnt)
rio read  <reg> <offset> <did>                - read <reg> register
rio write <reg> <offset> <did> <data>         - write <reg> register
rio dbell <did> <data> <n>                    - send doorbell n times
rio mem   <did> <offset> <length> <n>         - send data n times
rio msg   <did> <mbox> <length> <n>           - send message n times
                                  n=pos transmits n times
                                  n=neg retransmits n times
```

6.3 U-Boot Start Up Screen

```
U-Boot 1.3.4-03792-gf556483-dirty (Sep 15 2008 - 21:21:30)

CPU:   AMCC PowerPC 460GT Rev. A at 800 MHz (PLB=200, OPB=100,
EBC=100 MHz)
      Security/Kasumi support
      Bootstrap Option B - Boot ROM Location EBC (16 bits)
      32 kB I-Cache 32 kB D-Cache
Board: Arches - AMCC PPC460GT Reference Design
I2C:   ready
DTT:   1 is 25 C
DRAM:  512 MB (ECC not enabled, 400 MHz, CL3)
FLASH: 32 MB
Net:   ppc_4xx_eth0, ppc_4xx_eth1, ppc_4xx_eth2


Type run flash_nfs to mount root filesystem over NFS


Hit any key to stop autoboot:  0
```

Figure 8 - U-Boot Start Up Screen

6.4 U-Boot Environment Settings

```
bootcmd=run flash_self
bootdelay=5
baudrate=115200
loads_echo=
preboot=setenv ethact ppc_4xx_eth1;echo;echo Type "run flash_nfs"
to mount root filesystem over NFS;echo;
hostname=arches
netdev=eth1
nfsargs=setenv bootargs root=/dev/nfs rw
nfsroot=${serverip}:${rootpath}
ramargs=setenv bootargs root=/dev/ram rw
addip=setenv bootargs ${bootargs}
ip=${ipaddr}:${serverip}:${gatewayip}:${netmask}:${hostname}:${netd
ev}:off panic=1
addtty=setenv bootargs ${bootargs} console=ttyS0,${baudrate}
addmisc=setenv bootargs ${bootargs}
initrd_high=30000000
kernel_addr_r=400000
fdt_addr_r=800000
ramdisk_addr_r=C00000
hostname=arches
rootpath=/opt/eldk/ppc_4xxFP
flash_self=run ramargs addip addtty addmisc;bootm ${kernel_addr}
${ramdisk_addr} ${fdt_addr}
flash_nfs=run nfsargs addip addtty addmisc;bootm ${kernel_addr} -
${fdt_addr}
net_nfs=tftp ${kernel_addr_r} ${bootfile}; tftp ${fdt_addr_r} ${fdt_file};
run nfsargs addip addtty addmisc;bootm ${kernel_addr_r} -
${fdt_addr_r}
net_self_load=tftp ${kernel_addr_r} ${bootfile};tftp ${fdt_addr_r}
${fdt_file};tftp ${ramdisk_addr_r} ${ramdisk_file};
net_self=run net_self_load;run ramargs addip addtty addmisc;bootm
${kernel_addr_r} ${ramdisk_addr_r} ${fdt_addr_r}
load=tftp 200000 arches/u-boot.bin
update=protect off 0xFFFA0000 FFFFFFFF;era 0xFFFA0000
FFFFFFFF;cp.b ${fileaddr} 0xFFFA0000 ${filesize};setenv
filesize;saveenv
upd=run load update
kernel_addr=fe000000
fdt_addr=fe1e0000
ramdisk_addr=fe200000
pciconfighost=1
pcie_mode=RP:RP
```

U-Boot Environment Settings (continued)

ethaddr=00:1e:59:1f:00:a0
eth1addr=00:1e:59:1f:00:a1
eth2addr=00:1e:59:1f:00:a2
serverip=10.10.0.121
ipaddr=10.10.0.252
bootfile=uImage uImage
ramdisk_file=uRamdisk
fdt_file=arches.dtb
stdin=serial
stdout=serial
stderr=serial
ver=U-Boot 1.3.4-03792-gf556483-dirty (Sep 15 2008 - 21:21:30)
ethact=ppc_4xx_eth1

Environment size: 1788/16379 bytes

Appendix A – Memory Map

Board specific memory map, see table 4-1 of the Rigel/Calypso, Engineering Architecture Specification Rev. 1.01 – 10/12/07 for complete memory map information.

The following memory map is only an example; refer to the PowerPC 460GT user manual for specific memory configurations, many of the memory map setting are user defined.

A.1 - Chip Selects

Chip Select	Function	Address	Size	Description
PerCS0#	NOR FLASH	0xFC00_0000	64MB	X16 BUS, EBC
PerCS1#	FPGA	0x4000_0000	2MB	X16 BUS, EBC
PerCS2#	-	-	-	-
BankSel0#	DDR2	0x0000_0000	512MB	X64 BUS, DDR2

A.2 - Maximum Configuration

32 Bit Address		Function	CS#	Size/Reserved
Start	End			
0x0000_0000	0x1FFF_FFFF	DDR SDRAM	DDR CS0#	512MB/512MB
0x8000_0000	0x0003_FFFF	SRAM (L2 Cache)	-	32KB/256KB
0xF000_0000	0xFFFF_FFFF	BOOT Space EBC	PBus* CS0#	256MB/256MB
0x8000_0000	0x801F_FFFF	FPGA	PBus* CS2#	2MB/2MB
0xEF40_0000	0xEF40_003F	Local Configuration Registers		64B/1KB
0xEF60_0300	0xEF60_0307	UART0		8B/†16KB
0xEF60_0700	0xEF60_071F	IIC0		32B/†16KB
0xEF60_0900	0xEF60_0906	SPI		7B/†16KB
0xEF60_0E00	0xEF60_0EFF	Enet 0		256B/†16KB
0xEF60_0F00	0xEF60_0FFF	Enet 1		256B/†16KB
		PCI-x & SRIO		
		PCI-x Interrupt		

Notes: * PBus is Peripheral Bus; † 16KB is total for these Functions.

A.3 - I2C Device Addresses

Bus Description	Address	Connected	MFG. Device PN
MMC to AMC	TBD	U14 to AMC Connector	Microchip PIC24FJ64GA002
MMC to CPLD	TBD	U14 to U18	
CPU0, I2C0	A8	U1 to U33	Atmel AT24C32
CPU0, I2C0	94	U1 to U35	ADI AD7414-0
CPU1, I2C0	A8	U2 to U34	Atmel AT24C32
CPU1, I2C0	94	U2 to U36	ADI AD7414-0

Appendix B – FPGA Registers

Register 0 PCB Major Revision

Base + 0x0_0000

Bit #	PCB revision used with reg 1, example is rev 01.00 Where 01 is read in reg 0 and 00 is read in reg 1	Reset Value	Bit Type
0-7	Upper byte of Board ID	0x01	Read Only

Register 1 PCB Minor Revision

Base + 0x1_0001

Bit #	PCB revision used with reg 0, example is rev 01.00 Where as 01 is read in reg 0 and 00 is read in reg 1	Reset Value	Bit Type
0-7	Lower byte of Board ID	0x00	Read Only

Register 2 FPGA Revision

Base + 0x2_0002

Bit #	FPGA revision	Reset Value	Bit Type
0-7	FPGA revision information	0x01	Read Only

Register 3 CPU Active

Base +0x3_0003

Bit #	FPGA Revision	Reset Value	Bit Type
0-7	Only read by one CPU	0x01	Read Only

Register 4 Clock Enable / TMRCLK Divider

Base +0x4_0004

Bit #	Description		Reset Value	Bit Type
0	CLK_GMCRefClk_EN	1 = enabled	0	Read/Write
1	CLK_TMRCK_EN	1 = enabled	0	Read/Write
2	Reserved		0	Read/Write
3	Reserved		0	Read/Write
4	Reserved		0	Read/Write
5-7	TMRCLK div, based on 4 input (50MHz) 000 = divide by 1 001 = divide by 2 010 = divide by 4 011 = divide by 6 100 = divide by 10 101 = divide by 20 110 = divide by 40 111 = divide by 50		000 001 010 011 100 101 110 111	Read/Write

Register 5 FLASH Write Protect

Base +0x5_0005

Bit #	Description		Reset Value	Bit Type
0	FLASH0_WPn	1 = enabled	1	Read/Write
1	FLASH0_RY_BYN MASK bit	0 = hold low 1 = pass through to CPU	1	Read/Write
2	Reserved		1	Read/Write
3	Reserved		1	Read/Write
4	FLASH1_WPn	1 = enabled	1	Read/Write
5	FLASH1_RY_Byn MASK bit	0 = hold low 1 = pass through to CPU	1	Read/Write
6	Reserved		1	Read/Write
7	Reserved		1	Read/Write

Appendix C – CPLD Registers

Register 0 CPLD Revision

Base +0x00

Bit #	Description	Reset Value	Bit Type
0-7	CPLD revision information	0x01	Read Only

Register 1 Boot Mode Configuration Switch Settings

Base +0x01

Bit #	Description				Reset Value	Bit Type
0-2	Reserved				xxx	
3	Reserved				0000	Read Only
4-7	CFG_sw0…3, switch position 1 = CFG_SW0 = MSB					Read Only
	CPU	Boot Opt	SYSCLK	PCIe CLK		
	1 & 2	E	66.6MHz	100MHz	0000	
	1 & 2	B	66.6MHz	100MHz	0001	
	1 & 2	G	66.6MHz	100MHz	0010	
	1 & 2	E	66.6MHz	125MHz	0011	
	1 & 2	B	66.6MHz	125MHz	0100	
	1 & 2	G	66.6MHz	125MHz	0101	
	Reserved				0110	
	Reserved				0111	
	1 & 2	G	74.9MHz	100MHz	1000	
	1 & 2	G	83.3MHz	100MHz	1001	
	1 & 2	G	99.9MHz	100MHz	1010	
	Reserved				1011	
	1 & 2	G	74.9MHz	125MHz	1100	
	1 & 2	G	83.3MHz	125MHz	1101	
	1 & 2	G	83.3MHz	125MHz	1110	
	1 & 2	G	99.9MHz	125MHz	1111	

Register 2 Geographical Address

Base +0x2

Bit #	AMC Geograhical address bits 0…2 2 bits for every address bit 00 = low 01 = high 11 = 3-state	Reset Value	Bit Type
0	Reserved	x	Read Only
1	Reserved	0	Read Only
2-3	Bit 0	xx	Read Only
4-5	Bit 1	00	Read Only
6-7	Bit 2	xx	Read Only

Register 3 Power On Reset High
Base +0x03

Bit #	In AMC mode the MMC must write 0x55 before the board will boot When running, if the MMC writes 0xAA to register 04 and 0x55 to register 5 the board will reset and be in a powered down state	Reset Value	Bit Type
0-7	CPLD POR = 0xAA, reg. 4 & 5 need to be set	0x55	Read/Write

Register 4 Power On Reset Low
Base +0x04

Bit #	In AMC mode the MMC must write 0xAA before the board will boot When running, if the MMC writes 0xAA to register 4 and 0x55 to register 5 the board will reset and be in a powered down state	Reset Value	Bit Type
0-7	CPLD POR = 0x55, reg. 4 & 5 need to be set	0xAA	Read/Write

Register 5 Ethernet Control
Base +0x05

Bit #	Controls which IRQ is used for the listed IRQs 0 = IRQ0_CPU0, 1 = IRQ1_CPU0	Reset Value	Bit Type
0	ETH0_RESET, Active Low	0	Read/Write
1	ETH1_RESET, Active Low	0	Read/Write
2	ETH2_RESET, Active Low	0	Read/Write
3	ETH3_RESET, Active Low	0	Read/Write
4	ETH0_RESET, Active High	1	Read/Write
5	ETH1_RESET, Active High	1	Read/Write
6	ETH2_RESET, Active High	1	Read/Write
7	ETH3_RESET, Active High	1	Read/Write

Register 6 UART Control, EEPROM WP, SYSERR
Base +0x06

Bit #	Controls which IRQ is used for the listed IRQs 0 = IRQ_CPU0, 1 = IRQ1_CPU0	Reset Value	Bit Type
0	CPU0_FPRCEPFF 0 = shut off	1	Read/Write
1	CPU0_EEP_WP 1 = Write Protect	0	Read/Write
2	Reserved	0	Read/Write
3	SYSERR_CPU0	0	Read Only
4	CPU1_FORCEOFF 0 = shut off	1	Read/Write
5	CPU1_EEP_WP 1 = Write Protect	0	Read/Write
6	Reserved	0	Read/Write
7	SYSERR_CPU1	0	Read Only

Register 7 LED Control

Base +0x07

Bit #	LEDs are Bi-Color and controlled in pairs	Reset Value	Bit Type
0	LED 2, LED_UESER 1, Green	0	Read/Write
1	LED 2, LED_UESER 2, Yellow	1	Read/Write
2	LED 3, LED_UESER 3, Green	1	Read/Write
3	LED 3, LED_UESER 4, Yellow	0	Read/Write
4	User override for USER_LEDS 0 = default = status information LEDS changing colors, heart beat 1 = User control	0	Read/Write
5	User override for STATUS LEDS 0 = default = status information RED = reset, else green 1 = User control	0	Read/Write
6	LED_STATUS2, Red	1	Read Only
7	LED_STATUS3, Green	0	Read Only

Register 8 to F Reserved

Base +0x0x (where xx is registers 8 to F

Bit #	Bit Description	Reset Value	Bit Type
0	Reserved	0	Read/Write
1	Reserved	0	Read/Write
2	Reserved	0	Read/Write
3	Reserved	0	Read/Write
4	Reserved	0	Read/Write
5	Reserved	0	Read/Write
6	Reserved	0	Read/Write
7	Reserved	0	Read/Write

Appendix D – AMC Pin Assignments

Existing SRIO Ports per AMC.4				
Disputed Ports between AMC.4 & SCOPE				
Scope Ports per V 1.0				
PIN	**SIGNAL**	**DRIVEN BY**	**MATING PIN**	**FUNCTION ON AMC**
1	GND		First	Logic Ground
2	PWR	Carrier	First	Payload Power
3	PS1#	AMC L	Last	Presence 1
4	MP	Carrier	Second	Management Power
5	GA0	Carrier	Second	Geographical Address 0
6	Reserved		Second	Reserved
7	GND		First	Logic Ground
8	Reserved		Second	Reserved
9	PWR	Carrier	First	Payload Power
10	GND		First	Logic Ground
11	TX0+	AMC	Third	Port 0 Transmitter+
12	TX0-	AMC	Third	Port 0 Transmitter-
13	GND		First	Logic Ground
14	RX0+	Carrier	Third	Port 0 Receiver+
15	RX0-	Carrier	Third	Port 0 Receiver-
16	GND		First	Logic Ground
17	GA1	Carrier	Second	Geographical Address 1
18	PWR	Carrier	First	Payload Power
19	GND		First	Logic Ground
20	TX1+	AMC	Third	Port 1 Transmitter+
21	TX1-	AMC	Third	Port 1 Transmitter-
22	GND		First	Logic Ground
23	RX1+	Carrier	Third	Port 1 Receiver+
24	RX1-	Carrier	Third	Port Receiver1-
25	GND		First	Logic Ground
26	GA2	Carrier	Second	Geographical Address 2
27	PWR	Carrier	First	Payload Power
28	GND		First	Logic Ground
29	TX2+	AMC	Third	Port 2 Transmitter+
30	TX2-	AMC	Third	Port 2 Transmitter-
31	GND		First	Logic Ground
32	RX2+	Carrier	Third	Port 2 Receiver+
33	RX2-	Carrier	Third	Port 2 Receiver-
34	GND		First	Logic Ground

Appendix D – AMC Pin Assignments (continued)

Existing SRIO Ports per AMC.4				
Disputed Ports between AMC.4 & SCOPE				
Scope Ports per V 1.0				
PIN	**SIGNAL**	**DRIVEN BY**	**MATING PIN**	**FUNCTION ON AMC**
35	TX3+	AMC	Third	Port 3 Transmitter+
36	TX3-	AMC	Third	Port 3 Transmitter-
37	GND		First	Logic Ground
38	RX3+	Carrier	Third	Port 3 Receiver+
39	RX3-	Carrier	Third	Port 3 Receiver-
40	GND		First	Logic Ground
41	ENABLE#	Carrier	Second	Carrier Second AMC Enable
42	PWR	Carrier	First	Payload Power
43	GND		First	Logic Ground
44	TX4+	AMC	Third	Port 4 Transmitter+
45	TX4-	AMC	Third	Port 4 Transmitter-
46	GND		First	Logic Ground
47	RX4+	Carrier	Third	Port 4 Receiver+
48	RX4-	Carrier	Third	Port 4 Receiver-
49	GND		First	Logic Ground
50	TX5+	AMC	Third	Port 5 Transmitter+
51	TX5-	AMC	Third	Port 5 Transmitter-
52	GND		First	Logic Ground
53	RX5+	Carrier	Third	Port 5 Receiver+
54	RX5-	Carrier	Third	Port 5 Receiver-
55	GND		First	Logic Ground
56	SCL_L	IPMI Agent	Second	IPMB-L Clock
57	PWR	Carrier	First	Payload Power
58	GND		First	Logic Ground
59	TX6+	AMC	Third	Port 6 Transmitter+
60	TX6-	AMC	Third	Port 6 Transmitter-
61	GND		First	Logic Ground
62	RX6+	Carrier	Third	Port 6 Receiver+
63	RX6-	Carrier	Third	Port 6 Receiver-
64	GND		First	Logic Ground
65	TX7+	AMC	Third	Port 7 Transmitter+
66	TX7-	AMC	Third	Port 7 Transmitter-
67	GND		First	Logic Ground
68	RX7+	Carrier	Third	Port 7 Receiver+

Appendix D – AMC Pin Assignments (continued)

Existing SRIO Ports per AMC.4				
Disputed Ports between AMC.4 & SCOPE				
Scope Ports per V 1.0				
PIN	**SIGNAL**	**DRIVEN BY**	**MATING PIN**	**FUNCTION ON AMC**
69	RX7-	Carrier	Third	Port 7 Receiver-
70	GND		First	Logic Ground
71	SDA_L	IPMI Ag	Second	IPMB-L Data
72	PWR	Carrier	First	Payload Power
73	GND		First	Logic Ground
74	CLK1+	CLK1 Drive	Third	Synchronization Clock 1+
75	CLK1-	CLK1 Drive	Third	Synchronization Clock 1-
76	GND		First	Logic Ground
77	CLK2+	CLK2 Drive	Third	Synchronization Clock 2+
78	CLK2-	CLK2 Drive	Third	Synchronization Clock 2-
79	GND		First	Logic Ground
80	CLK3+	CLK1 Drive	Third	Synchronization Clock 3+
81	CLK3-	CLK1 Drive	Third	Synchronization Clock 3-
82	GND		First	Logic Ground
83	PS0#	AMC	Last	Presence 0
84	PWR	Carrier	First	Payload Power
85	GND		First	Logic Ground
86	GND		First	Logic Ground
87	RX8-	Carrier	Third	Port 8 Receiver-
88	RX8+	Carrier	Third	Port 8 Receiver+
89	GND		First	Logic Ground
90	TX8-	AMC	Third	Port 8 Transmitter-
91	TX8+	AMC	Third	Port 8 Transmitter+
92	GND		First	Logic Ground
93	RX9-	Carrier	Third	Port 9 Receiver-
94	RX9+	Carrier	Third	Port 9 Receiver+
95	GND		First	Logic Ground
96	TX9-	AMC	Third	Port 9 Transmitter-
97	TX9+	AMC	Third	Port 9 Transmitter+
98	GND		First	Logic Ground
99	RX10-	Carrier	Third	Port 10 Receiver-
100	RX10+	Carrier	Third	Port 10 Receiver+
101	GND		First	Logic Ground
102	TX10-	AMC	Third	Port 10 Transmitter-

Appendix D – AMC Pin Assignments (continued)

Existing SRIO Ports per AMC.4				
Disputed Ports between AMC.4 & SCOPE				
Scope Ports per V 1.0				
PIN	**SIGNAL**	**DRIVEN BY**	**MATING PIN**	**FUNCTION ON AMC**
103	TX10+	AMC	Third	Port 10 Transmitter+
104	GND		First	Logic Ground
105	RX11-	Carrier	Third	Port 11 Receiver-
106	RX11+	Carrier	Third	Port 11 Receiver+
107	GND		First	Logic Ground
108	TX11-	AMC	Third	Port 11 Transmitter
109	TX11+	AMC	Third	Port 11 Transmitter+
110	GND		First	Logic Ground
111	RX12-	Carrier	Third	Port 12 Receiver-
112	RX12+	Carrier	Third	Port 12 Receiver+
113	GND		First	Logic Ground
114	TX12-	AMC	Third	Port 12 Transmitter -
115	TX12+	AMC	Third	Port 12 Transmitter +
116	GND		First	Logic Ground
117	RX13-	Carrier	Third	Port 13 Receiver-
118	RX13+	Carrier	Third	Port 13 Receiver+
119	GND		First	Logic Ground
120	TX13-	AMC	Third	Port 13 Transmitter -
121	TX13+	AMC	Third	Port 13 Transmitter +
122	GND		First	Logic Ground
123	RX14-	Carrier	Third	Port 14 Receiver-
124	RX14+	Carrier	Third	Port 14 Receiver+
125	GND		First	Logic Ground
126	TX14-	AMC	Third	Port 14 Transmitter -
127	TX14+	AMC	Third	Port 14 Transmitter +
128	GND		First	Logic Ground
129	RX15-	Carrier	Third	Port 15 Receiver-
130	RX15+	Carrier	Third	Port 15 Receiver+
131	GND		First	Logic Ground
132	TX15-	AMC Thi	Third	Port 15 Transmitter -
133	TX15+	AMC Thi	Third	Port 15 Transmitter +
134	GND		First	Logic Ground
135	RX16-	Carrier Thi	Third	Port 16 Receiver-
136	RX16+	Carrier Thi	Third	Port 16 Receiver+

Appendix D – AMC Pin Assignments (continued)

Existing SRIO Ports per AMC.4				
Disputed Ports between AMC.4 & SCOPE				
Scope Ports per V 1.0				
PIN	**SIGNAL**	**DRIVEN BY**	**MATING PIN**	**FUNCTION ON AMC**
137	GND		First	Logic Ground
138	TX16-	AMC Por	Port	Port 16 Transmitter -
139	TX16+	AMC Thi	Third	Port 16 Transmitter +
140	GND		First	Logic Ground
141	RX17-	Carrier	Port	Port 17 Receiver-
142	RX17+	Carrier	Third	Port 17 Receiver+
143	GND		First	Logic Ground
144	TX17-	AMC	Third	Port 17 Transmitter-
145	TX17+	AMC	Third	Port 17 Transmitter +
146	GND		First	Logic Ground
147	RX18-	Carrier	Third	Port 18 Receiver-
148	RX18+	Carrier	Third	Port 18 Receiver+
149	GND		First	Logic Ground
150	TX18-	AMC	Third	Port 18 Transmitter -
151	TX18+	AMC	Third	Port 18 Transmitter +
152	GND		First	Logic Ground
153	RX19-	Carrier	Third	Port 19 Receiver-
154	RX19+	Carrier	Third	Port 19 Receiver+
155	GND		First	Logic Ground
156	TX19-	AMC	Third	Port 19 Transmitter-
157	TX19+	AMC	Third	Port 19 Transmitter +
158	GND		First	Logic Ground
159	RX20-	Carrier	Third	Port 20 Receiver-
160	RX20+	Carrier	Third	Port 20 Receiver+
161	GND		First	Logic Ground
162	TX20-	AMC	Third	Port 20 Transmitter-
163	TX20+	AMC	Third	Port 20 Transmitter +
164	GND		First	Logic Ground
165	TCLK	Carrier	Second	JTAG Test Clock
166	TMS	Carrier	Second	JTAG Test Mode Select
167	TRST#	Carrier	Second	JTAG Test reset
168	TDO	AMC	Second	JTAG Test data Output
169	TDI	Carrier	Second	JTAG Test data Input
170	GND		First	Logic Ground

Appendix E – General Pin Assignments

J1 – Ethernet Connector CPU0

Pin No	Description
1	X0_MDI_0p
2	X0_MDI_0n
3	X0_MDI_1p
4	X0_MDI_2p
5	X0_MDI_2n
6	X0_MDI_1n
7	X0_MDI_3p
8	X0_MDI_3n
9	ETH0_STATUS0
10	ETHO_STATUS1
11	ETHO_LOS
12	+2.5v
13	CHASSIS GND
14	CHASSIS GND
15	NC
16	NC

J2 – Ethernet Connector CPU1

Pin No	Description
1	X2_MDI_0p
2	X2_MDI_0n
3	X2_MDI_1p
4	X2_MDI_2p
5	X2_MDI_2n
6	X2_MDI_1n
7	X2_MDI_3p
8	X2_MDI_3n
9	ETH2_STATUS0
10	ETH2_STATUS1
11	ETH2_LOS
12	+2.5v
13	CHASSIS GND
14	CHASSIS GND
15	NC
16	NC

P1 – COP Trace Header

Pin No	Description
1	COP TD0
2	NC
3	COP TDI
4	COP TRST#
5	NC
6	COP VCC
7	COP_TCK
8	NC
9	COP TMS
10	NC
11	COP HALT#
12	NC
13	NC
14	NC
15	NC
16	GND

P2 – COP Trace CPU Selector Header

Pin No	Description
1	CPU0 HALT#
2	COP_HALT#
3	COP_HALT#
4	CPU1 HALT#

P3 – RS232 Selector Header

Pin No	Description
1	RS232 CTL
2	GND

P4 – CPLD Connector

Pin No	Description
1	CPLD TCK
2	GRD
3	CPLD TDO
4	3.3V_STDBY
5	CPLD TMS
6	NC
7	NC
8	NC
9	CPLD_TDI
10	GND

P5 – Altera Byteblaster Connector

Pin No	Description
1	DCLK
2	GND
3	CONF_DONE
4	+3.3V
5	nCONFIG
6	nCE
7	DATA0
8	FLASH_nCE_IO_VB1N0_D2
9	DATA1_IO_VB1N0_C1
10	GND

P6 – RS232 Connector for CPU0

Pin No	Description
1	CPU0_RS232_TX
2	CPU0_RS232_RX
3	GND
4	CPU0_RS232_RTS
5	CPU0_RS232_CTS
6	GND
7	CHASSIS GRD
8	CHASSIS GRD

P7 – RS232 Connector for CPU1

Pin No	Description
1	CPU1_RS232_TX
2	CPU1_RS232_RX
3	GND
4	CPU1_RS232_RTS
5	CPU1_RS232_CTS
6	GND
7	CHASSIS GRD
8	CHASSIS GRD

PWR1 – Barrel Power Connector

Pin No	Description
1	+12V
2	GND
3	STD_ALONE_ENABLE (NC/PIN2)
4	+12V

Acronyms and Abbreviations

AMC ..... Advanced Mezzanine Card
ATCA .....AdvancedTCA
BMC ..... Baseboard Management Controller
COP……..……Co-processor
GMII…..... Gigabit Media Independent Interface
I2C, I^2C, IIC……. .....Intelligent Interface Controller
IPMB ..... Intelligent Platform Management Bus
IPMI.....Intelligent Platform Management Interface
JTAG..... Joint Test Advisory Group
MMC…….……..Module Management Controller
PETs .....Platform Event Traps
PMC .....peripheral management controller
POST ..... Power-On Self Test
RGMII.....Reduced Gigabit Media Independent Interface
SOL..... Serial-Over-LAN
SSIF ..... SMbus System Interface
RAS.....Reliability, Availability and Serviceability
RMCP .....Remote Management Control Protocol
TSEC .....Triple Speed Ethernet Controller